



Programming for Your New MINOS VARC

An example of register values

N. Felt

28 March 2001

Note: Τησ δοχυμεντ ισ υνδερ χονστρυχτιον ανδ υνδεργοεσ φρεθυεντ χηανγεσ. Πλεασε χονσυлт τη λατεστ περσιον.

1. Initialization

When the VA RC is powered up there are certain registers that must be set to a predetermined value.

1.1. VARC Initialization

e is the ETC number (0-5)

bb is the board address

0xbb3e 0018	<=	0x00	--set etc to disabled
0xbb3e 0040	<=	0x07	--set etc to enable all chips on FEB 0
0xbb3e 0048	<=	0x28	--hold delay
0xbb3e 0050	<=	0x04	--VA clock period
0xbb40 0018	<=	0x24	--enable 50ms buffer switching, put in Sparsifier in run mode

1.2. Slow-Control

Before writing to each register you must check to see the Slo-controls are not busy. This can be done by reading the status register. For VFB0 bias settings the DAC addresses are as follows:

0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0000	<=	0X50	--Vfp
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0004	<=	0X02	--Vfs
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0008	<=	0X7D	--Sha-bias
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 000C	<=	0X30	--Trigger threshold
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0010	<=	0X40	--Calibration charge
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0018	<=	0X00	--Turn all voltages on

This would be the settings for VFB1

0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0020	<=	0X50	--Vfp
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0024	<=	0X02	--Vfs
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0028	<=	0X7D	--Sha-bias
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 002C	<=	0X30	--Trigger threshold
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0030	<=	0X40	--Calibration charge
0xbb20 01E0	=>		--bit 1 indicates it is busy, wait until this is 0
0xbb20 0038	<=	0X00	--Turn all voltages on

After the board is initialized the VARC is then in disabled mode. From this mode the board can then be set to Cal-inj mode, Triggerless-pedestal mode, Run mode or a test mode.

2. Disabled Mode

0xbb3e 0018	and	1111 1100	--set etc to disabled
0xbb40 0018	and	1111 1110	-- turn Sparsifier test bit off
0xbb40 0018	or	0000 0100	-- take Sparsifier out of test mode

3. Cal-inject mode

3.1. Single Shot

0xbb3e 0018	and	1111 0101	--set ETC (0-5) to Cal-inject
0xbb3e 0018	or	0000 0001	--set ETC (0-5) to Cal- inject
0xbb40 0018	and	1111 1110	-- turn Sparsifier test bit off
0xbb40 0018	or	0000 0100	-- put Sparsifier in run mode

While in single-shot mode every time a word is written to the CAL_TRIG_XQT_CTRL register one 64-bit data word will be produced.

For example to cal-inject FEB 0, chip 1 channel 6 while in this mode we would simply write to the following register.

0xbb30 0038	<=	0x01 06	--do cal-inject to specified chip
-------------	----	---------	-----------------------------------

If the register is written too frequently the charge injected will be reduced. This can be avoided with a delay of at least 1ms between cal-injects.

3.2. Automatic

0xbb3e 0018	and	1111 1101	--set ETC (0-5) to Auto Cal-inject
0xbb3e 0018	or	0000 1001	--set ETC (0-5) to Auto Cal-inject
0xbb40 0018	and	1111 1110	-- turn Sparsifier test bit off
0xbb40 0018	or	0000 0100	-- put Sparsifier in run mode

While the VARC is in automatic Cal-inject mode, it will produce one 64-bit data word for every XQT signal it sees. If the XQT signal is too frequent the charge injected will be reduced. This can be avoided with an XQT period > 1ms.

The VARC automatically cycles through all enabled chips. The situation where a chip is enabled but not installed will create errors. Chips are enabled using the CHIP_TRIG_ENABLE_REGISTER

4. Triggerless-pedestal mode

In Triggerless-pedestal mode the chips are read out in the same manner as if a trigger was actually received from the FEB. Therefore, the data can be read from the VME buffers in the same manner as in Run mode. Each Triggerless-pedestal command has the potential of creating large amounts of data. Care must be taken to avoid overflowing the VME readout buffer.

4.1. Single Shot

0xbb3e 0018	and	1111 1110	--set ETC (0-5) to Triggerless-pedestal
0xbb3e 0018	or	0000 0010	--set ETC (0-5) to Triggerless-pedestal
0xbb40 0018	and	1111 1110	-- turn Sparsifier test bit off
0xbb40 0018	or	0000 0100	-- put Sparsifier in run mode

While in single-shot mode sparsification is automatically disabled; therefore, every time a word is written to the CAL_TRIG_XQT_CTRL register 22 64-bit data word will be produced for each chip.

For example to read the pedestals of FEB 0, chip 0 chip1 and chip3, while in this mode we would simply write to the following register.

```
0xbb30 0038    <=    0x00 07    --do cal-inject to specified chip
```

Since we are reading the pedestals from 3 chips we will expect to see 66 64-bit data words in the output buffer.

4.2. Automatic

```
0xbb3e 0018    and    1111 1110    --set ETC (0-5) to Auto Triggerless-pedestal
0xbb3e 0018    or     0000 1010    --set ETC (0-5) to Auto Triggerless-pedestal
0xbb40 0018    and    1111 1110    -- turn Sparsifier test bit off
0xbb40 0018    or     0000 0100    -- put Sparsifier in run mode
```

While the VARC is in automatic mode, it will produce 22 64-bit data words for every XQT signal it sees. The VARC automatically cycles through all enabled chips. Chips are enabled using the CHIP_TRIG_ENABLE_REGISTER

5. Run Mode

In this mode the VARC will be processing triggers and filling the output buffers.

```
0xbb3e 0018    or     0000 0011    --set ETC (0-5) to run
0xbb40 0018    and    1111 1110    -- turn Sparsifier test bit off
0xbb40 0018    or     0000 0100    -- take Sparsifier out of test mode
```

6. Buffer Readout

Reading data from the VME buffers is the same for any mode the VARC is in. The contents of the buffers can be read using the external 50ms buffer switch or it can manually switch the buffers using a VME command.

6.1. Manual VME method

This method gives the user control of the buffer switching, thus, control of the speed the data is read out. To use this readout method, bit 5 in the Sparsifier control register must be set to a 0 BEFORE data is written to the buffers. This should be done when changing running modes.

```
0xbb40 0018    and    1101 1111    --manual VME select
```

While in Manual VME readout mode, every time the buffer is switched, by flipping bit 6, the amount of data and valid buffer can be read from BUFFER_LEVEL_U and BUFFER_LEVEL_L.

```
0xbb40 0018    xor     0100 0000    -- switch buffer

0xbb40 0038    -- read lower byte of amount of data words available.
0xbb40 0040    -- read upper amount of data words available and from what buffer
```

Read correct amount of data from the buffers

6.2. 50 ms Interrupt method

To use this readout method, bit 5 in the Sparsifier control register must be set to a 1 BEFORE data is written to the buffers. This should be done when changing running modes.

0**xb**40 0018 or 0010 0000 --manual VME select

The Interrupt method is similar to the Manual VME method except the buffers switch automatically every 50 ms. When the buffers switch an interrupt will be generated and the amount of data and valid buffer can be read from BUFFER_LEVEL_U and BUFFER_LEVEL_L.

0**xb**40 0038 -- read lower byte of amount of data words available.
0**xb**40 0040 -- read upper amount of data words available and from what buffer

Read correct amount of data from the buffers

7. Setting Pedestal and Threshold RAM

7.1. Pedestals

Once the pedestal and threshold has been determined for each channel of each chip the value must be written to the memory on the VARC. The format of the RAM address bits is as follows:

Bit13
0 Select Threshold
1 Select Pedestal

Bit[12:10]
Select ETC 0-5 (1 of 6)

Bit9
Select VFB 0-1 (1 of 2)

Bit[8:7]
Select VA chip 0-2 (1 of 3)

Bit[6:2]
Select VA Channel 0-21 (1 of 22)

Bit1
Always 0

Bit0
Always 0

For example to write the Pedestal value of 0x0A for channel 6, chip 1, VFB0, ETC 0 you would write to the following register:

0**xb**60 2098 <= 0X0A

To write the Threshold value of 0x05 for channel 6, chip 1, VFB0, ETC 0 you would write to the following register:

0**xb**60 0098 <= 0X05

Since this memory is not dual-ported you will not be able to write to this memory while VA chip data is being processed. This means during run mode, cal-inject operations or triggerless-pedestal operations. You can only write to this memory when the memory is not otherwise busy. The memory is not busy in disabled mode. The memory is also not busy when in triggerless-pedestal mode when not processing triggerless-pedestals.

8. How do I know if my VARC is working?

If you suspect your VARC might not be properly working there are a few functional checks you can do without using a test-stand.

8.1. Error Registers

The error registers in each FPGA can give clues to what is going wrong. If there is an error in any FPGA it will be reported to the Master Clock. By checking this status bit you can find out if an error has been produced. You can find out the specific error and clear the error by reading each error register in all FPGAs.

0**xb**10 0020 If bit 1 is set there is an error. You need to read ETC and Sparsifier errors

0**xb**3e 0060 Check for ETC errors

0**xb**40 0060 Check for Sparsifier errors

The specific errors that correspond to each bit can be found in the users manual.

8.2. Buffer Test Pattern

Bit # 6 in Sparsifier control register selects which buffer the test pattern will be written to. writing 0 selects buffer A and writing a 1 selects buffer B. Bit # 2 selects if the Sparsifier is in test mode or normal mode. Writing 0 in bit 2 will put the Sparsifier in test mode. Writing a 1 to bit 0 will begin the process of writing to the buffers. This bit must be cleared out with a 0 before the test pattern can be written again.

0 xb 40 0018	and	1001 1010	-- put Sparsifier in test mode and select buffer A
0 xb 40 0020	= >		-- check to see that the buffers are not busy
0 xb 40 0018	or	0000 0001	-- initiate the writing of the test pattern to buffer A

0 xb 40 0018	and	1101 1010	-- put Sparsifier in test mode and clear bit 0
0 xb 40 0018	or	0100 0000	-- select buffer B
0 xb 40 0020	= >		-- check to see that the buffers are not busy
0 xb 40 0018	or	0000 0001	-- initiate the writing of the test pattern to buffer B

The VME buffer memory should now contain the buffer test pattern and can be read when the buffers are not busy being written into.

8.3. Data flow from ETC to VME buffer

If you think the data in your buffer is not correct you can tell the ETC to generate a known set of ADC data values. The ETC will produce ADC values that are equal to the channel numbers. Channel 1 will have an ADC value of 1. Channel 2 will have an ADC value of 2. This is with the exception of channel 0 which will hold the common-mode-correction value.

0**xb**30 0018 or 0000 0100 -- Turn on fake data

When you are finished with this mode you must turn off this bit

0x**bb**30 0018 and 1111 1011 -- Turn off fake data

It does not matter when this bit is turned on or off. The timing doesn't matter. You can use this fact to test the VARC when you think it is stuck in a mode that is creating errors. If you ever see suspicious data being produced by the VARC you would simply turn on this bit and a known set of data should be produced. If the data is other than the expected then there is something wrong with VARC.

One test we commonly do is with the triggerless-pedestal mode. In this mode we do not need a FEB or VMM. You do a triggerless-pedestal run on any set of chips using the ETC to generate the data. This is a good way to determine everything is working properly with the VARC and everything downstream. You should see 22 channels for each chip you trigger with sparsification, pedestal subtraction and common-mode-correction off. All ADC values will be equal to the channel number.

If the previous test seems to be working you can connect the VMM with the FEB disconnected. When the ETC data generation is off you should see somewhat constant ADC values with noise ~ 2 ADC counts. This is with the exception of channel 0 which is the common-mode-correction value.

8.4. Cal-injects

Cal-injects are also a nice way of testing the electronics including the FEB. Please see the section on Cal_injects.